UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/786,771	02/25/2004	Stephan R. Yhann	07844-634001	2862

21876 7590 05/04/2006

FISH & RICHARDSON P.C.
P.O. Box 1022
MINNEAPOLIS, MN 55440-1022

EXAMINER	
WASHBURN, DANIEL C	
ART UNIT	**PAPER NUMBER**
2628	

DATE MAILED: 05/04/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

PTO-90C (Rev. 10/03)

Office Action Summary	Application No.	Applicant(s)	
	10/786,771	YHANN ET AL.	
	Examiner	**Art Unit**	
	Dan Washburn	2628	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *21 February 2006*.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-26* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-26* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *25 February 2004* is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)

2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date _____.

4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .

5) ☐ Notice of Informal Patent Application (PTO-152)

6) ☐ Other: _____.

DETAILED ACTION

Response to Arguments

Applicant's arguments with respect to claims 1-26 have been considered but are moot in view of the new ground(s) of rejection.

Claim Rejections - 35 USC § 103

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

Claims 1-5, 7, 8, 11-14, 16, 18-24, and 26 are rejected under 35 U.S.C. 103(a) as being unpatentable over Yoshino et al. (US 6,141,462) in view of Hollinger (US 5,715,331).

As to claims 1 and 22, Yoshino describes a method and a computer program product, tangibly stored on a computer-readable medium, for processing graphical elements, the product comprising instructions operable to cause a programmable system to: receive an original graphical element having an associated original type; blend at least part of the original graphical element and at least part of at least one other graphical element to produce a transformed graphical element having an associated transformed type, the transformed type being different than the original type; store information about the original type for the original graphical element; and process at least one of the transformed graphical element and an adjacent graphical element using the stored information about the original type. For example, Yoshino offers Figure 2,

which illustrates the general steps involved in processing an image. The computer program has access to a PostScript file, which has a data list of stored information pertaining to all the graphical objects contained within the PostScript file. The stored information includes the size and shape of each object and the color of each object, among other details column 5 lines 7-29. Figure 2C illustrates the program evaluating the interaction among objects to decide if any objects should be blended together. The program creates a bit mapped image of the objects and generates an adjoining relationship list of image parts and saves this information for later use. The program then decides which objects should be blended together and it combines certain objects to create a single blended object column 5 lines 21-48. Once the bit map is generated the program moves on to recognizing blended objects, storing the blended objects and any original objects in a data list, and processing the bit mapped image using a trapping process. The trapping process uses the stored information about the original objects including the type of each object, the color of each object, and how each object is blended with any surrounding objects, to determine how to best trap the recently created bit mapped image column 7 lines 58-67 and column 8 lines 1-8. Objects P1 through P7 of Figure 2 originated in a PostScript file, the graphical elements P1 through P6 are blended together to create a single blended object. If object P1 is considered the original graphical element and object P2 is considered one other graphical element then the original graphical element P1 is blended with another graphical element P2 (along with P3 through P6) to create a new graphical element. The blended graphical element is transformed from an individual object type to a blended object type. The

transformed blended object type is considered different than the original individual object type. The original object information remains stored in the PostScript file for later use and the program processes, traps, to be specific, the blended transformed graphical element (P1 through P6) and an adjacent graphical element (P7) using the spatial relationship information of the original images that is collected right after the bit map is created and before the images are blended Figure 2. Figure 1 offers a typical computer system that houses the described program. The program is contained within RAM 28, which is a computer-readable medium. Yoshino doesn't describe that a rule associated with processing of graphical elements having the original type differs from a corresponding rule associated with processing of graphical elements having the transformed type.

However, Hollinger describes a method and system of receiving an original graphical element having an original type (column 2 lines 12-15, in this case the type is a raster image of an object); blending at least part of the original graphical element and at least part of one other graphical element to produce a transformed graphical element having an associated transformed type, the transformed type being different than the original type (column 2 lines 17-41 describes performing edge detection on a raster image and then creating a vector image of all the edges that make up the raster image. The vector image is then combined with the raster image to create a composite raster-vector image, which is considered a transformed type that is different than the original type); and processing at least one of the transformed graphical element and an adjacent graphical element, wherein a rule associated with processing of graphical elements

having the original type differs from a corresponding rule associated with processing of graphical elements having the transformed type (column 2 lines 29-35 describes that an objective of the disclosed invention is to create a composite image that can be easily manipulated, rotated, and scaled. The processing algorithms, or rules, applied to a composite raster-vector image during image manipulations such as rotating and scaling are drastically different from the processing algorithms, or rules, applied to a raster image during the same manipulations, therefore the composite image, which is considered an image having a transformed type, will have very different processing rules applied to it when compared to the processing rules applied to the original raster image, which is considered an image having an original type.) It would have been obvious to one of ordinary skill in the art at the time of the invention to include in Yoshino the system and method of creating a composite raster-vector image, as taught by Hollinger, in order to create an image type that is easy to manipulate and gives a more accurate representation of the real-world printed image.

With regard to claim 2, Yoshino includes storing information about a type associated with the at least one other graphical element. For example, Yoshino describes that all information about all graphical elements P1 through P7, of Figure 2, including color, shape, and that each object is or is not blended with another object, is stored in a PostScript file which the program has full access to column 5 lines 7-20.

Concerning claim 3, Yoshino describes storing information about a colorspace and a color for the original graphical element. For example, Yoshino offers Figure 3, which illustrates color information pertaining to each graphical object. Parts P1 through

P6 are colored with varying densities of cyan, and part P7 is colored with yellow. The stored colorspace in this case is the CMYK colorspace. The program uses this stored colorspace along with the density of each color component (C, M, Y, and K) associated with each graphical element to determine which graphical elements should be blended together column 5 lines 7-20 and lines 52-67.

Regarding claims 4 and 5, Yoshino describes storing an original shape of the at least part of the original graphical element, wherein storing the original shape includes storing the original shape as a path of the at least part of the original graphical element. For example, Yoshino describes that all the information describing the original data is stored in a PostScript file. The PostScript file stores all characteristics associated with each graphical element, including color and shape. The images are stored as vector images, as is common in a PostScript file, which means the original shape of each graphical element is stored as a path of vectors that defines the graphical element column 5 lines 7-14.

As to claim 7, Yoshino describes a method wherein processing includes locating one or more edges in the transformed graphical element using the stored original shape. For example, Figure 3 illustrates the process of trapping two graphical elements using stored information about the original shape. It compares this process to the process of trapping two graphical elements without using stored information about the original shape. Specifically, Yoshino describes using the stored information about how the original graphical elements were combined to locate only the outer edges of the blended shape in the bit-mapped image. Once the trapping process has the information

describing the blended image it can more effectively trap P7 with blended object P1 through P6 column 7 lines 58-67 and column 8 lines 1-8. Figure 3E shows trapping the bit-mapped image using the stored information about the original shape to locate the edges of the blended transformed graphical element, and Figure 3F shows trapping the bit-mapped image without the blending information.

With regard to claims 8 and 23, Yoshino discloses a method and product wherein the transformed graphical element is a rasterized representation of the blended at least part of the original graphical element and at least part of the at least one other graphical element. For example, Yoshino offers Figure 6, which illustrates taking in vector images and bit mapping them to transform the vector images into raster images. Referring back to Figure 3, if object parts P1 through P7 are bit mapped then the transformed graphical element (blended elements P1 through P6) is a rasterized representation of the blended at least part of the original graphical element (P1, for example) and at least part of the at least one other graphical element (P7, for example, contained within the box that P1 though P6 creates) column 8 lines 18-56.

Concerning claim 11 and 24, Yoshino describes a method and a product with instructions operable to cause a programmable system to process, wherein processing includes trapping at least one of the transformed graphical element and the adjacent graphical element. For example, Yoshino offers Figure 5, which illustrates trapping transformed graphical element P1 through P6 with graphical element P7 column 6 lines 56-67 and column 7 lines 1-6.

Regarding claim 12, Yoshino describes a method wherein trapping includes using a path of the transformed graphical element to represent a path of the at least part of the original graphical element. For example, Yoshino offers Figure 5A, which describes using the path of the blended raster image P1 through P6 that is adjacent to the perimeter of P7 as a reference for the trapping process.

As to claim 13, Yoshino includes a method wherein trapping includes using a color of the transformed graphical element to calculate a color of a trap element. For example, Yoshino offers Figure 5, which describes trapping area RR in Figure 5A and trapping sub areas RR2 through RR6 in Figure 5B. Trapping areas RR2 through RR6 use the color of the transformed graphical elements P2 through P7 to calculate the color of the trap element for each area. In this particular case yellow is lighter than cyan so the yellow color within P7 is spread out into P2 through P6. The original cyan color contained within RR2 through RR6 doesn't change, but a yellow component is added to each of RR2 through RR6 to create a trapped image.

With regard to claim 14, Yoshino describes a method wherein trapping includes using trapping rules that depend on the stored information about the original type. For example, Yoshino describes that the trapping process is modified based on the stored information about the similar color and position of the original objects P1 through P7 of Figure 3. Specifically, a blending process takes place before the trapping process that decides if any or all of the objects P1 through P7 should be blended together. The criterion for blending objects together is based on the color of each original object from the PostScript file. If the objects from the PostScript file are adjacent to each other and

their colors are similar (based on a threshold value of 3%) then the two objects are blended together. The trapping process treats blended objects as one single object, and therefore the trapping rules between a blended set of objects and a separate object are different than the trapping rules between an unblended set of objects and a separate object. Whether or not objects are blended is based on the stored color information associated with the original unblended objects column 7 lines 58-67 and column 8 lines 1-8.

Concerning claim 16, Yoshino describes a method wherein blending includes flattening at least part of the original graphical element and at least part of at least one other graphical element. The original graphical element is considered to be any one of the unblended elements P1 through P7 described in Figure 3. When the original graphical elements P1 through P6 are blended together they are merged to become one object column 5 lines 21-29. The process of merging objects together on one plane is considered a flattening process.

As to claim 18, Yoshino describes a method wherein the original graphical element was produced by blending two or more previous graphical elements; and storing information about the original type includes storing information about a type associated with at least one of the previous graphical elements. For example, Yoshino describes that the original graphical elements are image data in the form of a PostScript file. The PostScript file comprises vector images and these vector images can be made up of vector strokes or combinations of vector strokes and vector fills. The graphical elements may overlap and these graphical elements are considered graphical elements

that are produced by blending two or more previous graphical elements. The PostScript file stores information about the type associated with the graphical element, in this case, whether the element is a blend of vector strokes and vector fills, or just consists of vector strokes, and whether image is a blended image or an unblended image column 5 lines 7-14.

Regarding claim 19, Yoshino describes a method wherein the original type comprises a member of a set of types, the types in the set of types including raster, vector stroke, vector fill, image mask, soft mask, glyph, and gradient. For example, Yoshino describes that the original image data is in the form of a PostScript file. PostScript files contain information in the form of vector images. The image types comprise individual objects consisting of vector strokes and possibly vector fills and blended objects consisting of vector strokes and possibly vector fills, which means the object type is vector stroke or both vector stroke and vector fill column 5 lines 7-14 and Figure 3.

With regard to claim 20, Yoshino describes a method wherein the transformed type is raster. For example, Yoshino describes taking vector images from a PostScript file and bit mapping them to create a raster image of the vector images. From there the program transforms any individual objects that meet the blending criteria into one or more blended objects to create a blended object that is also a raster image. In this case objects P1 through P6 are transformed into a blended object type that is a raster image column 5 lines 7-29.

Concerning claims 21 and 26, Yoshino describes a method and product wherein the original type is not associated with the transformed graphical element. For example, Yoshino describes taking individual objects that consist of vectors and transforming them into a blended object that is a raster, or bit mapped, image column 5 lines 7-48. The blended object type is not associated with the individual object type and the raster image is not associated with the vector images, to the extent possible when a new type is created from the transformation of an original type.

Claims 6, 9, 17, and 25 are rejected under 35 U.S.C. 103(a) as being unpatentable over Yoshino et al. (US 6,141,462) in view of Hollinger (US 5,715,331) and further in view of Ahlstrom et al. (US 6,594,030).

Regarding claim 6, Yoshino describes storing an original shape as part of the original graphical element, as described in the rejection of claim 4. Yoshino doesn't describe that the original shape is stored as a text glyph of the original graphical element.

However, Ahlstrom describes a method of trapping objects on a page that includes generating a minimal amount of traps for any given publication column 2 lines 56-63. Ahlstrom also describes that text objects, which are considered text glyphs, can be blended with background objects and then the text object and background object can be trapped column 4 lines 42-50. It would have been obvious to one of ordinary skill in the art at the time of the invention to include in Yoshino the method of storing the original shape as a text glyph of the original graphical object in order to give the

blending and trapping program the added capability of creating graphic designs that include text.

As to claims 9, 17, and 25, Yoshino describes storing information about the original type of an object in a graphical element, where the graphical element is an opaque graphical element. For example, Yoshino offers Figure 2B, which is a data list of opaque graphical elements that contains information about the original type of each graphical element. Yoshino doesn't teach that the information is stored in an invisible graphical element or that the original graphical element is transparent and the transformed graphical element is opaque.

However, Ahlstrom describes that the original type of a graphical element is stored in an invisible graphical element and that the original graphical element can be transparent but the transformed graphical element is opaque. For example, Ahlstrom describes a transparent graphical element, which may be completely transparent, or invisible, and can potentially contain text. The text containing transparent graphical element is trapped against background elements underneath it to create a transformed opaque graphical element column 4 lines 42-50. If the user chooses to exclude text from the transparent graphical element, or chooses to make the text transparent, then the user has created a completely transparent, or invisible, graphical element. It would have been obvious to one of ordinary skill in the art at the time of the invention to include in Yoshino the invisible graphical element and transparent graphical element that is transformed into an opaque graphical element in order to allow the user to add text to a publication that appears to be completely independent of all other graphical

elements on the page and combine transparent elements with opaque graphical elements to create a combination opaque element that has in the foreground an at least partially transparent graphical element, such as a text string, and in the background an opaque element, such as a background pattern or color. The advantage of creating opaque elements out of transparent elements with opaque elements underneath is that a user can transform an image with many layers into an image with one layer, which is easier to work with and saves as a smaller file.

Claim 10 is rejected under 35 U.S.C. 103(a) as being unpatentable over Yoshino et al. (US 6,141,462) in view of Hollinger (US 5,715,331) and further in view of Uemura et al. (US 2003/0214534).

Regarding claim 10, Yoshino describes storing information about the original type of an object in a graphical element, where the graphical element is an opaque graphical element. For example, Yoshino offers Figure 2B, which is a data list of opaque graphical elements that contains information about the original type of each graphical element. Yoshino doesn't teach that the information is stored in an XML element.

However, Uemura describes storing information using XML tags to store data items from all different application programs in a uniform format paragraph 0012. It would have been obvious to one of ordinary skill in the art at the time of the invention to include in Yoshino the ability to store information about the original type of an image in XML format as taught by Uemura to allow the stored data to be readable by many different programs. The advantage of saving image information, including type, size, color, and position on the screen, in an XML format is that the created images can be

viewed using a wide range of software, including those that aren't compatible with PostScript.

Claim 15 is rejected under 35 U.S.C. 103(a) as being unpatentable over Yoshino et al (US 6,141,462) in view of Hollinger (US 5,715,331) and further in view of Gupta et al. (US 2004/0141194).

Concerning claim 15, Yoshino includes processing at least one of the transformed graphical element and the adjacent graphical element, as described in the rejection of claim 1. Yoshino doesn't describe that the processing includes halftoning at least one of the transformed graphical element and the adjacent graphical element.

However, Gupta describes halftoning at least one of the transformed graphical element and an adjacent graphical element. For example, Gupta describes using a color digital halftoning technique to improve the quality of printed images while reducing the amount of ink used minimizing misregistration of colors on a page. Misregistration leads to gaps between colors printed on a page, which deteriorates the overall quality of any images printed. It would have been obvious to one of ordinary skill in the art at the time of the invention to include in Yoshino the halftoning process as described by Gupta in order to increase the quality of images created by Yoshino's software program by halftoning the images in conjunction with Yoshino's trapping process.

Conclusion

The prior art made of record and not relied upon is considered pertinent to applicant's disclosure. Speck (US 6,654,145) describes applying various trapping rules to an image.

Applicant's amendment necessitated the new ground(s) of rejection presented in this Office action. Accordingly, **THIS ACTION IS MADE FINAL.** See MPEP § 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the date of this final action.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Dan Washburn whose telephone number is (571) 272-5551. The examiner can normally be reached on Monday through Friday 8:30 a.m. to 5:00 p.m..

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Ulka Chauhan can be reached on (571) 272-7782. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

DW

DW

4/27/06

UChauhan

ULKA CHAUHAN
SUPERVISORY PATENT EXAMINER